

Machine Learning Approaches for Breast Cancer Diagnosis and their Comparison

*Thesis submitted in
partial fulfillment of the requirements
for the degree of*

Bachelor of Technology

in

Computer Science and Engineering

by

M Anvesh

[110CS0514]

under the guidance of

Dr. B. Majhi



Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela, Odisha, 769 008, India
May 2014



Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India.

Dr. B. Majhi
Professor

May 10, 2014

Certificate

This is to certify that the work in the thesis entitled *Machine Learning Approaches for Breast Cancer Diagnosis and their Comparison* by *M.Anvesh* is a record of an original research work carried out under my supervision and guidance in partial fulfilment of the requirement for the award of the degree of *Bachelor of Technology in Computer Science and Engineering*. Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

Dr. B. Majhi

Acknowledgement

I take this opportunity to express my profound gratitude and deep regards to my guide Dr. B. Majhi for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. He motivated and inspired me through the entire duration of work, without which this project could not have seen the light of the day.

I convey my regards to all the faculty members of the Department of Computer Science and Engineering, NIT Rourkela for their valuable guidance and advices at appropriate times. I would like to thank my friends for their help and assistance all through this project.

Last but not the least, I express my profound gratitude to the Almighty and my parents for their blessings and support without which this task could have never been accomplished.

M.Anvesh

Abstract

Machine learning approaches are used for building systems that can solve various diagnostic problems. Since breast cancer is highly incident on women, there is a need for such systems. Mammograms are used for early detection of breast cancer. The breast cancer diagnostic system extracts features from these mammograms and classifies them as malignant or benign. These systems are very helpful to doctors in detecting and diagnosing the disease faster than any other traditional methods.

In this thesis an attempt has been made to classify the extracted features from mammograms as benign or malignant by using Naive Bayes, K-NN, Multilayer Perceptron, Radial Basis Function Networks, Support Vector Machine approaches. Performance variation of the approaches by varying various parameters is studied. Finally the results are compared to find the best performing approaches.

Contents

Certificate	i
Acknowledgement	ii
Abstract	iii
List of Figures	vi
List of Tables	vii
1 Introduction	1
1.1 Breast Cancer Diagnosis System	1
1.2 Motivation	1
1.3 Objectives and Scope of Work	2
1.4 Outline of the Thesis	3
2 Naive Bayes classification	4
2.1 Probabilistic model	4
2.2 Naive Bayes for Continuous Data	5
2.2.1 Normal distribution	5
2.2.2 Kernel Density Estimation	5
3 K-Nearest Neighbor Classification	6
4 Multilayer Perceptron	7
4.1 Introduction	7
4.2 Network Architecture	7
4.3 Back Propagation Algorithm	8

5	Radial Basis Function Networks	10
5.1	Introduction	10
5.2	Network Architecture	11
5.3	Radial Basis Functions	12
5.4	Training	12
6	Support Vector Machines	13
6.1	Introduction	13
6.2	History	13
6.3	Linear Support Vector Machines	14
6.4	Nonlinear Support Vector Machine	16
7	Results	18
8	Conclusion and Future Work	25
	Bibliography	26

List of Figures

4.1	Network Architecture for Multilayer Perceptron	8
4.2	Usage of Sigmoid activation function	8
5.1	Network Architecture for Radial Basis Function Networks	11
7.1	Variation in accuracy for different values of k	19
7.2	Variation in accuracy for different network Architectures	20
7.3	Variation in accuracy for different network Architectures	21
7.4	Variation in accuracy for different values of Gaussian function variance	22
7.5	Comparison of the approaches used varying the training dataset sizes	24

List of Tables

7.1	Accuracies of various approaches for different training dataset sizes	23
-----	---	----

Chapter 1

Introduction

1.1 Breast Cancer Diagnosis System

Breast cancer is considered as one of the deadly diseases for women, but significant survival rates are possible with early detection of the cancer. Generally the diagnostic management of the breast cancer is a very difficult job. Mammography has been effectively used to screen women for breast cancer detection. The physicians analyze the mammographic images to predict the possibility of breast cancer. The physicians might not correctly predict the cancer due to the issues related with human fatigue and habituation [8]. Breast cancer diagnosis system is a software tool used to detect breast cancer by analyzing the mammographic images. Hence these systems are very useful for faster detection and diagnosis.

1.2 Motivation

Machine learning a sub-field of Artificial Intelligence is used to achieve thorough understanding of the learning process and to implant learning capabilities in computer system. It has various applications in the areas of science, engineering and the society. Machine learning approaches can provide generalized solutions for a wide range of problems effectively and efficiently. The machine learning approaches make computers more intelligent.

Machine learning helps in solving prognostic and diagnostic problems in a variety of medical domains [8]. It is mainly used for prediction of disease progression, for therapy planning, support and for overall patient management. Hypothesis from the patient data can be drawn from

expert systems mechanisms that use medical diagnostic reasoning [8]. As mentioned earlier breast cancer is dreadful, so there is a need for computerized systems that emulate the doctors expertise in detecting the disease and help in accurate diagnosis. Machine learning has various approaches for building such systems. There is no single approach for all the problems and each approach perform differently for different problems. So there is a need for finding the approaches that perform well for a particular problem. In this thesis various approaches are used for breast cancer diagnosis and they are compared to find the best performing ones.

1.3 Objectives and Scope of Work

The research was carried out with the following objectives

- (i) To study various machine learning approaches for breast cancer diagnosis through their implementation.
- (ii) To make a comparative study of the approaches.

For the purpose of research, I have considered only the classification task involved in such systems and used the existing feature space. The extraction of features from the mammographic images is not considered. The machine learning approaches that were considered here could be used for any other classification problem. I have focused mainly on breast cancer diagnosis, a medical domain problem.

1.4 Outline of the Thesis

This thesis consists of seven chapters following this chapter.

Chapter 2: Naive Bayes Classification

Naive Bayes classification for discrete and continuous data is discussed. Usage of Normal distribution and Kernel density estimations for continuous data are explained in this chapter.

Chapter 3: K Nearest Neighbor Classification

A basic level description of the K Nearest Neighbor approach for classification is explained in this chapter.

Chapter 4: Multilayer Perceptron

An introduction to multilayer perceptron and its network architecture is explained. Backpropagation algorithm for learning the network parameters is specified in this chapter.

Chapter 5: Radial Basis Function Networks

The details of radial basis functions, radial basis function networks and the network architecture are discussed. Incremental gradient descent algorithm and various approaches for learning the network parameters are specified in this chapter.

Chapter 6: Support Vector Machines

Linear and Nonlinear SVM's for classification are thoroughly discussed in this chapter.

Chapter 7: Results

The implementation results of various approaches are specified and their comparison details are discussed in this chapter.

Chapter 8: Conclusion and future work

This chapter discusses the outcome of the research work and future research directions.

Chapter 2

Naive Bayes classification

A Naive Bayes classifier is a simple probabilistic classifier based on Bayes theorem considering strong independence assumptions. A more descriptive term would be "independent feature model". Only small amount of data is required to estimate the parameters necessary for classification.

2.1 Probabilistic model

The classifier follows the conditional model $p(c \mid v_1, v_2, \dots, v_n)$ on the independent class variable c . Using Bayes' theorem we can write

$$p(c \mid v_1, v_2, \dots, v_n) = p(c \mid v_1, v_2, \dots, v_n) * \frac{p(v_1, v_2, \dots, v_n \mid c)}{p(v_1, v_2, \dots, v_n)}$$

The above equation can be written as

$$posterior = prior * \frac{likelihood}{evidence}$$

Using independence assumption, the conditional distribution over class c is given by

$$p(c \mid v_1, v_2, \dots, v_n) = p(c) \prod_1^n \frac{p(v_i \mid c)}{p(v_1, \dots, v_n)}$$

A Bayes classifier as a function classify is defined as follows

$$\text{Classify}(v_1, v_2, \dots, v_n) = \text{argmax}_c p(C = c) \prod_1^n p(v_i | C = c)$$

Model parameters are estimated using relative frequencies from training set.

Prior for a class = $\frac{\text{number of samples in the class}}{\text{total number of samples}}$

2.2 Naive Bayes for Continuous Data

2.2.1 Normal distribution

When the data is continuous we assume that the continuous values associated with each class are distributed according to normal distribution. Suppose the training data contains a continuous attribute x we first segment the data by class then compute the mean and variance of x in each class. Let μ be the mean of the values in x , σ^2 be variance for x associated with class c . Then the probability density $p(x = v | c)$ can be computed from the equation [2].

$$p(x = v | c) = \frac{1}{\sigma_c \sqrt{2\pi}} e^{\frac{-(v-\mu)^2}{2\sigma^2}}$$

2.2.2 Kernel Density Estimation

Kernel density estimation is a non-parametric way of estimating the probability density function. The probability $p(v_i | C = c)$ can be estimated using the following equation [1].

$$p(v_i | C = c) = \frac{1}{N_{ch}} \sum_{j=1}^{N_c} K(v_i, v_{j|i|c})$$

$$K(a, b) = \frac{1}{\sqrt{2\pi}} e^{\frac{-(a-b)^2}{2h^2}}$$

where K is a Gaussian function kernel with mean zero and variance one, N_c is the number of data points belonging to class c , $v_{j|i|c}$ is the feature value in the i^{th} position of the j^{th} input in class c and h is a constant called smoothing parameter.

Chapter 3

K-Nearest Neighbor Classification

In pattern recognition, K Nearest Neighbor algorithm is a non-parametric algorithm that can be used for classification and regression. It defers the decision to generalize beyond the training examples till a new query is encountered. The training examples are represented as vectors in a multidimensional feature space and each example is labeled by a class. During the training phase the feature vectors and their class labels are stored. While in the classification phase where k is a user defined constant and the new unlabeled vector is assigned a class that is more frequent among its k nearest neighbors. The distance is calculated by one of the following measures:

- Euclidean Distance
- Minkowski Distance
- Mahalanobis Distance

Chapter 4

Multilayer Perceptron

4.1 Introduction

A multilayer perceptron(MLP) is a feed forward artificial neural network model that maps the set of inputs to the appropriate outputs. To solve the non-linearly separable problems, a number of neurons are connected in layers to form a multilayer perceptron [3]. A single perceptron is sufficient to express linear decision surfaces but multilayer perceptron can express non-linear decision surfaces. Small linearly separable sections of the inputs are identified by each perceptron. Outputs of one layer of perceptrons are passed to another layer and finally combined into output layer of perceptrons to produce the results. The perceptrons uses various non-linear activation functions.

4.2 Network Architecture

A multilayer perceptron has one input layer, one output layer and any number of hidden layers. The neurons in the hidden layer uses nonlinear activation functions.

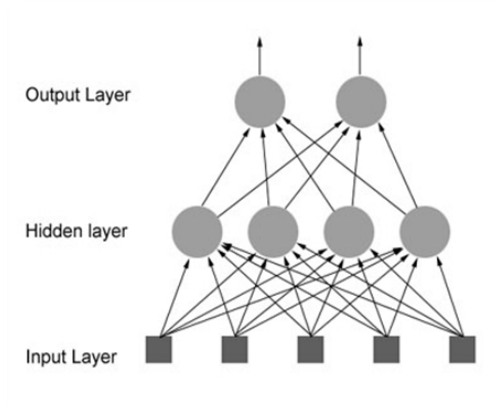


Figure 4.1: Network Architecture for Multilayer Perceptron

Sigmoid activation function is used

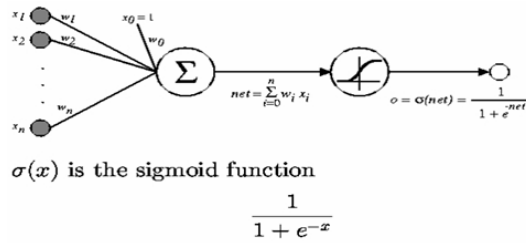


Figure 4.2: Usage of Sigmoid activation function

4.3 Back Propagation Algorithm

The back propagation algorithm learns the weights of a multilayer network, given a network with a fixed set of units and interconnections. It employs a gradient descent to minimize the squared error between the network output values and the target values for these outputs. We used only one output unit. The following algorithm is applied to a three layered network containing two layers of sigmoid units.

Algorithm 1 :Back Propagation Algorithm

- 1: Until satisfied, Do
- 2: For each training example, do
- 3: Input the training example to the network and compute the network outputs
- 4: For each output unit k

$$\delta_k \leftarrow o_k(1 - o_k)(t_k - o_k)$$

- 5: For each hidden unit h

$$\delta_h \leftarrow o_h(1 - o_h) \sum_{k \in \text{outputs}} w_{hk} \delta_k$$

- 6: Update each network weight w_{ij}

$$w_{ij} \leftarrow w_{ij} + \Delta w_{ij}$$

$$\text{Where } \Delta w_{ij} = \eta \delta_i x_{ij}$$

Notations:

- x_{ij} denotes the input from node i to unit j and w_{ij} denotes the corresponding weight.
- δ_n denotes the error term associated with unit n.
- t is the target output and o is the output from the network.

Chapter 5

Radial Basis Function Networks

5.1 Introduction

Radial basis function network is an artificial neural network which uses radial basis functions as activation functions. These networks are feed forward networks which can be trained using supervised training algorithms. These networks are used for function approximation in regression, classification and time series predictions [13]. Radial basis function networks are three layered networks where the input layer units does no processing, the hidden layer units implement a radial activation function and the output layer units implement a weighted sum of the hidden unit outputs. Nonlinearly separable data can easily be modeled by radial basis function networks. To use the radial basis function networks we have to specify the type of radial basis activation function, the number of units in the hidden layer and the algorithms for finding the parameters of the network.

5.2 Network Architecture

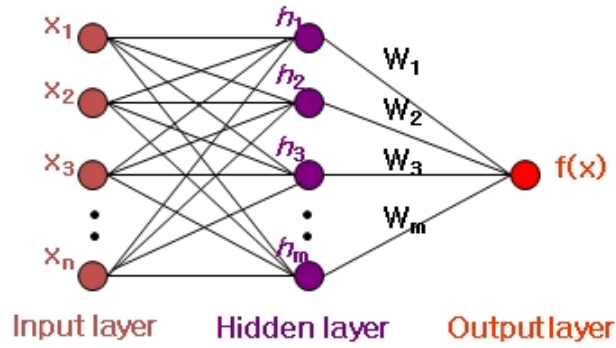


Figure 5.1: Network Architecture for Radial Basis Function Networks

The network has only three layers. When Gaussian radial basis function is used:

$$f(x) = \sum_i w_i h_i(x)$$

$$h_i(x) = e^{\frac{-(x-c_i)^2}{r^2}}$$

Where c_i is the centre of i_{th} hidden neuron

r is the width

$h(x)$ is the Gaussian function

x is the input

w_i is the weight for the connection between the i_{th} hidden unit and the output. Weights, width, centers form the parameters that has to be learned from the training data.

5.3 Radial Basis Functions

A function is radial basis if its output depends on the distance of input from the origin or from a given stored center.

5.4 Training

There are two levels of learning, in the first level we learn the centers and the width, in the second level we learn the weights for the connections between the hidden layer and the output. Different learning algorithms may be used for learning the radial basis function network parameters.

Learning the centers:

The centers can be learned by using the k-means algorithm.

Learning the width:

Width is chosen by normalization

$$r = \frac{\text{Maximum distance between any two centers}}{\sqrt{\text{Number of centers}}}$$

Learning the weights:

The weights can be learned by using the stochastic gradient descent algorithm.

Each training example is a pair of the form $\langle x, t \rangle$, where x represent the input values, t is the target output value and o is output from the network. η is the learning rate.

Algorithm 2 :Gradient Descent

- 1: Initialize each w_i to some small random value
- 2: Until the termination condition is met, Do
- 3: For each $\langle x, t \rangle$ in training example ,Do
- 4: Find the output o from the network
- 5: For each weight w_i , Do

$$\Delta w_i = \Delta w_i + \eta(t - o)x_i$$

$$w_i \leftarrow w_i + \Delta w_i$$

Chapter 6

Support Vector Machines

6.1 Introduction

In machine learning support vector machines are supervised learning models which contain various learning algorithms that analyze and recognize patterns used for classification. This is relatively new learning method for binary classification. It is a non-probabilistic binary linear classifier that builds a model for classifying the new data into one of the classes. The SVM's aim at finding the hyperplane that separates the data perfectly into two classes. When the data is non- separable, it can be mapped into higher dimensional feature space where the data becomes separable [5].

6.2 History

The original SVM algorithm was introduced by Vladimir N .Vapnik and colleagues. The first main paper seems to be (Vapnik,1995) and the earliest mention was in (Vapnik, 1979).

6.3 Linear Support Vector Machines

We are given n training examples $\{x_i, y_i\}$, $i = 1, 2, 3, \dots, n$ where each example has d dimensions ($x_i \in \mathbb{R}^d$) and a class label with one of two values ($y_i \in \{-1, 1\}$). The hyperplane that is parameterized by a vector (w), and constant b is expressed by the equation [4].

$$w \cdot x + b = 0$$

When such a hyperplane is given, the function which classifies the training data is given by

$$f(x) = \text{sign}(w \cdot x + b)$$

Assuming all data is at a distance larger than 1 from the hyperplane the following two constraints follow for the training set are

$$w^T x_i + b \geq 1 \text{ if } y_i = 1$$

$$w^T x_i + b \leq -1 \text{ if } y_i = -1$$

more compactly

$$y_i(w \cdot x_i + b) \geq 1 \quad \forall i$$

The geometric distance between the data point and the hyperplane is given by

$$d((w, b), x_i) = \frac{y_i(w \cdot x_i + b)}{\|w\|} \geq \frac{1}{\|w\|}$$

So the hyperplane that maximizes the geometric distance to the closest data points is needed. This can be accomplished by minimizing $\|w\|$ subject to distance constraints. The main method of doing this is with Lagrange multipliers [4].

The problem is transformed into

$$\text{minimize : } W(\alpha) = \sum \alpha_i - \frac{1}{2} \sum \sum \alpha_i \alpha_j y_i y_j x_i^T x_j$$

$$\text{subject to : } \sum \alpha_i y_i = 0$$

$$\text{and } 0 \leq \alpha_i \quad \forall \alpha_i$$

where α is the vector of n Lagrange multipliers to be determined and $i=1,2..n$
 $j=1,2..n$

A matrix $(H)_{ij} = y_i y_j (x_i \cdot x_j)$ is used for more compact notation [4]

$$\text{minimize : } W(\alpha) = -\alpha^T \mathbf{1} + \frac{1}{2} \alpha^T H \alpha$$

$$\text{subject to : } \alpha^T y = 0 \text{ and } 0 \leq \alpha$$

This optimization problem is known as a Quadratic Programming problem (QP). From the above equations the optimal hyperplane is given by the equation [4]

$$w = \sum \alpha_i y_i x_i$$

So the vector w is the linear combination of training examples. Interestingly it can be shown that

$$\alpha_i (y_i (x_i \cdot w + b) - 1) = 0$$

that is when $y_i (x_i \cdot w + b) > 1$ then $\alpha_i = 0$. Hence only the data points closest to the hyperplane contribute to w . The examples for which $\alpha_i > 0$ are called support vectors. These are the only ones needed for defining the the optimal hyperplane [4].

Assuming we have the optimal Φ we determine b to specify the hyperplane fully [4].

$$(w \cdot x^+ + b) = +1$$

$$(w \cdot x^- + b) = -1$$

Solving the equations we get $b = -\frac{1}{2} * (w \cdot x^+ + w \cdot x^-)$

6.4 Nonlinear Support Vector Machine

When the data is not linearly separable then we need to pre-process the data so that the original data can be mapped into some higher dimensional feature space. So we need to find a mapping $z = \Phi(x)$ that transforms the input vector to higher dimension. Given a mapping $z = \Phi(x)$, we set our quadratic programming problem by replacing all occurrences of x with $\Phi(x)$. The problem formulation is similar to that of linear SVM with small changes [4]

$$\text{minimize} : W(\alpha) = -\alpha^T \mathbf{1} + \frac{1}{2} \alpha^T H \alpha$$

$$(H)_{ij} = y_i y_j (\Phi(x_i) \cdot \Phi(x_j))$$

$$\text{Then } w = \sum \alpha_i y_i \Phi(x_i)$$

The classifier equation is [4]

$$\begin{aligned} f(x) &= \text{sign}(w \cdot \Phi(x) + b) \\ &= \text{sign}([\sum \alpha_i y_i \Phi(x_i)] \cdot \Phi(x) + b) \\ &= \text{sign}(\sum \alpha_i y_i (\Phi(x_i) \cdot \Phi(x)) + b) \end{aligned}$$

The concerns for the above specified procedure is choosing $\Phi()$ and when the data is transformed into an exponentially large dimension then the construction of matrix H that requires the dot product creates more burden. By increasing the complexity overfitting becomes a concern [4]. To avoid the above mentioned problems various kernels are used for implicit mapping of the original feature space to a higher dimensional space and represents the dot product in that higher dimensional feature space.

$$K(x_a, x_b) = \Phi(x_a) \cdot \Phi(x_b)$$

where there won't be any need for mapping $z = \Phi(x)$ explicitly.

The optimization formulation remains the same but the matrix $(H)_{ij} = y_i y_j (K(x_i, x_j))$.

The classifier equation is

$$f(x) = \text{sign}(\sum \alpha_i y_i K(x_i, x) + b)$$

$$b = y_{sv} - \sum_t \alpha_t y_t K(x_t, x_{sv})$$

sv represents any support vector and t represents set of all support vectors

The Gaussian radial basis function kernel is [4]

$$K(x_a, x_b) = e^{-\frac{\|x_a - x_b\|^2}{2\sigma^2}}$$

Chapter 7

Results

All the approaches discussed in the previous chapters are used to solve Breast cancer diagnosis classification problem. Based on the attributes which describe the characteristics of the cell nuclei, we have to classify a particular instance of data as benign or malignant.

Description of Dataset:

Breast cancer dataset is obtained from machine learning repository provided by the University of California, Irvine. The features are extracted from the mammographic images of breast mass.

They describe the characteristics of the cell nuclei present in the image.

Total Number of instances: 699

Number of Attributes: 9 plus the class attribute

Class distribution:

Benign : 458 (65.5)

Malignant : 241 (34.5)

The plots given below specify how the accuracies of the approaches used for classification varies depending on certain parameters.

Training dataset size is: 400

Test dataset size is: 299

Results

K-NN Classifier: The value of K is not fixed, a random value has to be selected.

The following plot shows how accuracy varies with k.

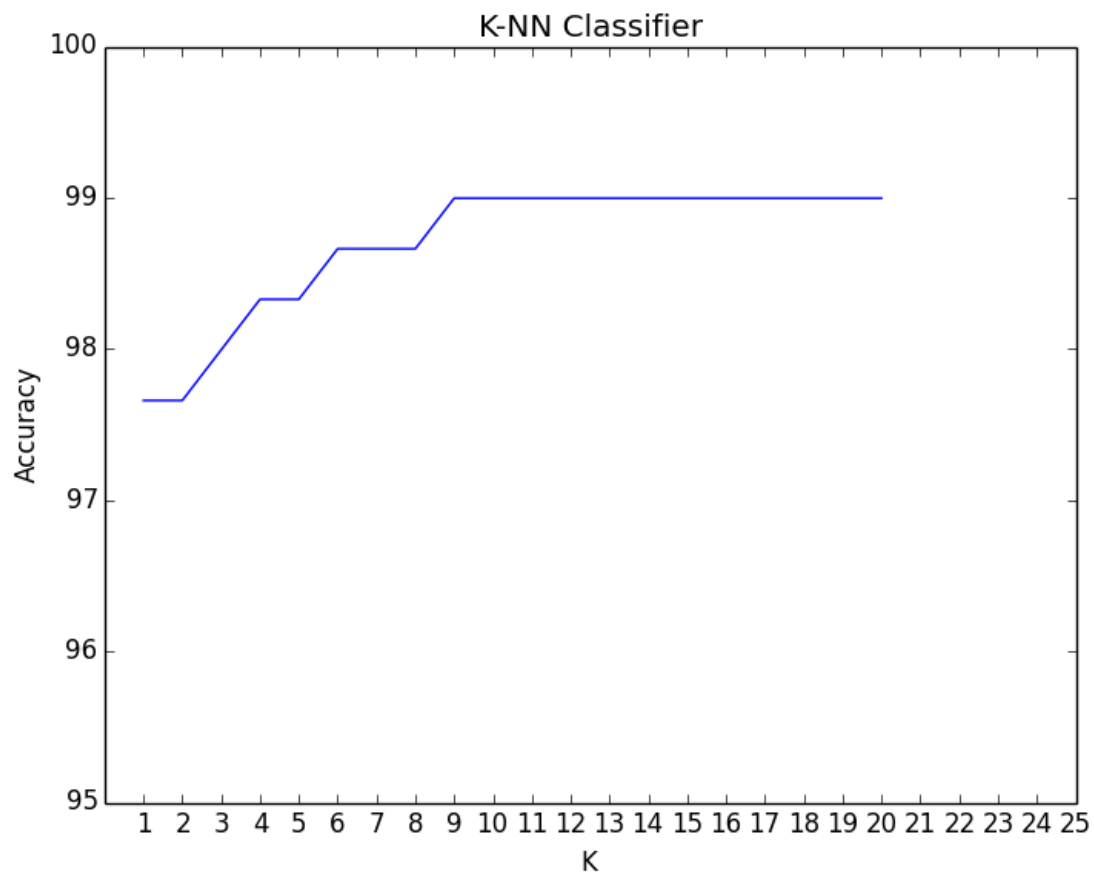


Figure 7.1: Variation in accuracy for different values of k

Results

Multilayer Perceptron: The number of hidden layer neurons which defines the network architecture effects the accuracy of the approach. The plot shows how accuracy varies with number of hidden layer neurons.



Figure 7.2: Variation in accuracy for different network Architectures

Radial Basis Function Networks: The number of hidden layer neurons effects the accuracy of approach. The plot shows how accuracy varies with number of hidden layer neurons.

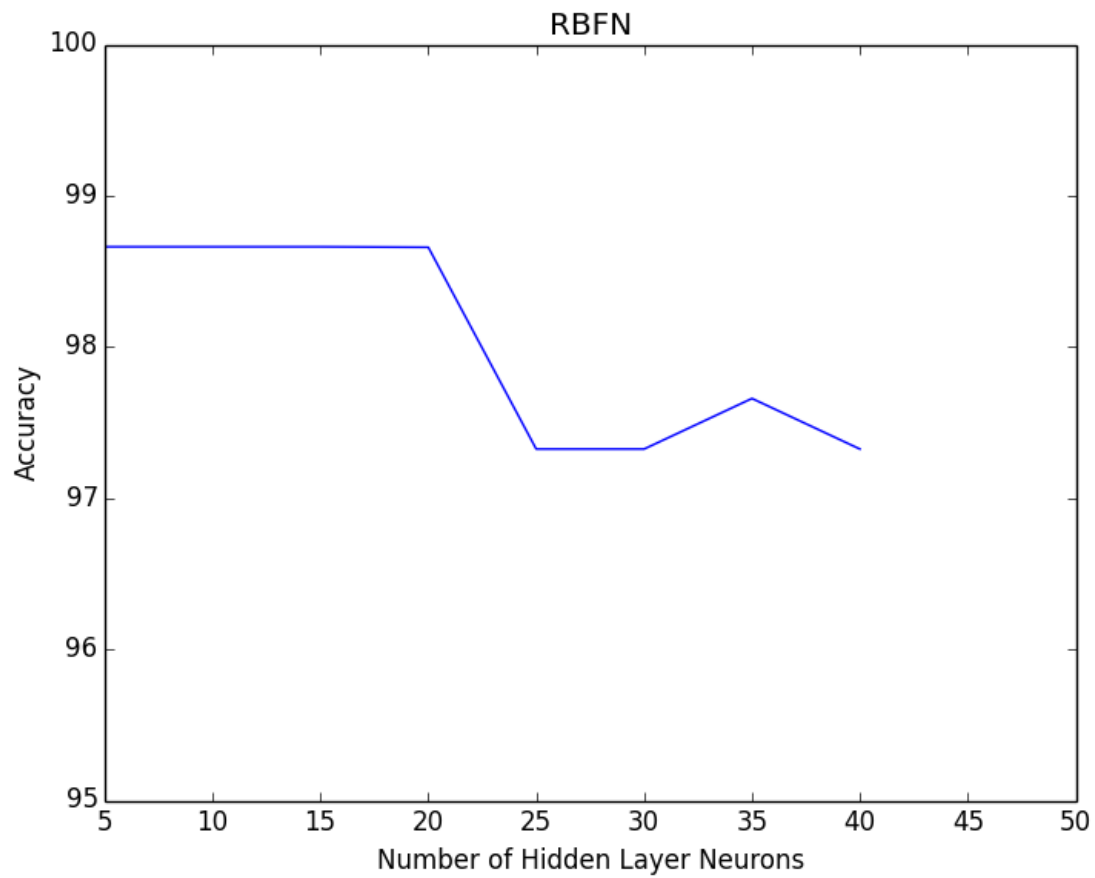


Figure 7.3: Variation in accuracy for different network Architectures

Results

Support Vector Machines: The plot shows how the accuracy varies with the variance of the Gaussian kernel.

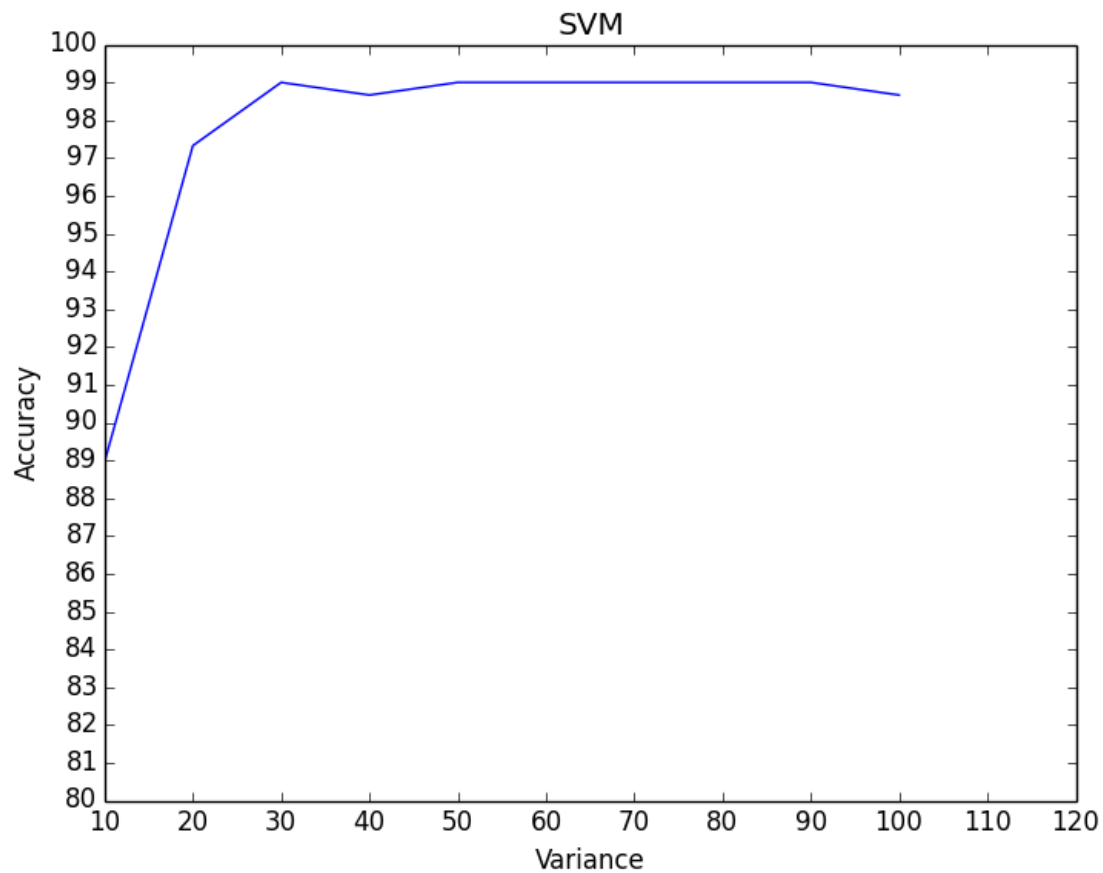


Figure 7.4: Variation in accuracy for different values of Gaussian function variance

Results

Accuracies of various approaches at different training dataset sizes are specified in the following table.

Table 7.1: Accuracies of various approaches for different training dataset sizes

Approach	Training Dataset Size				
	200	300	400	500	600
Naive Bayes Classifier (Normal Distribution)	68.5371	73.9348	76.5886	77.8894	78.7879
Naive Bayes Classifier (Kernel Density Estimation)	96.994	98.2456	98.9967	99.4975	98.9899
K-NN Classifier(K=10)	96.7936	97.9950	98.9967	100	100
Multilayer Perceptron (Hidden Layer units=25)	68.5317	73.9348	76.5886	78.8945	79.7980
Radial Basis Function Networks (Hidden Layer units=10)	97.3948	97.995	98.6622	100	98.9899
Support Vector Machine (RBF Kernel, Variance=30)	97.3948	97.9950	98.9967	98.4925	97.9798

The following plot shows a comparison between all the approaches used.

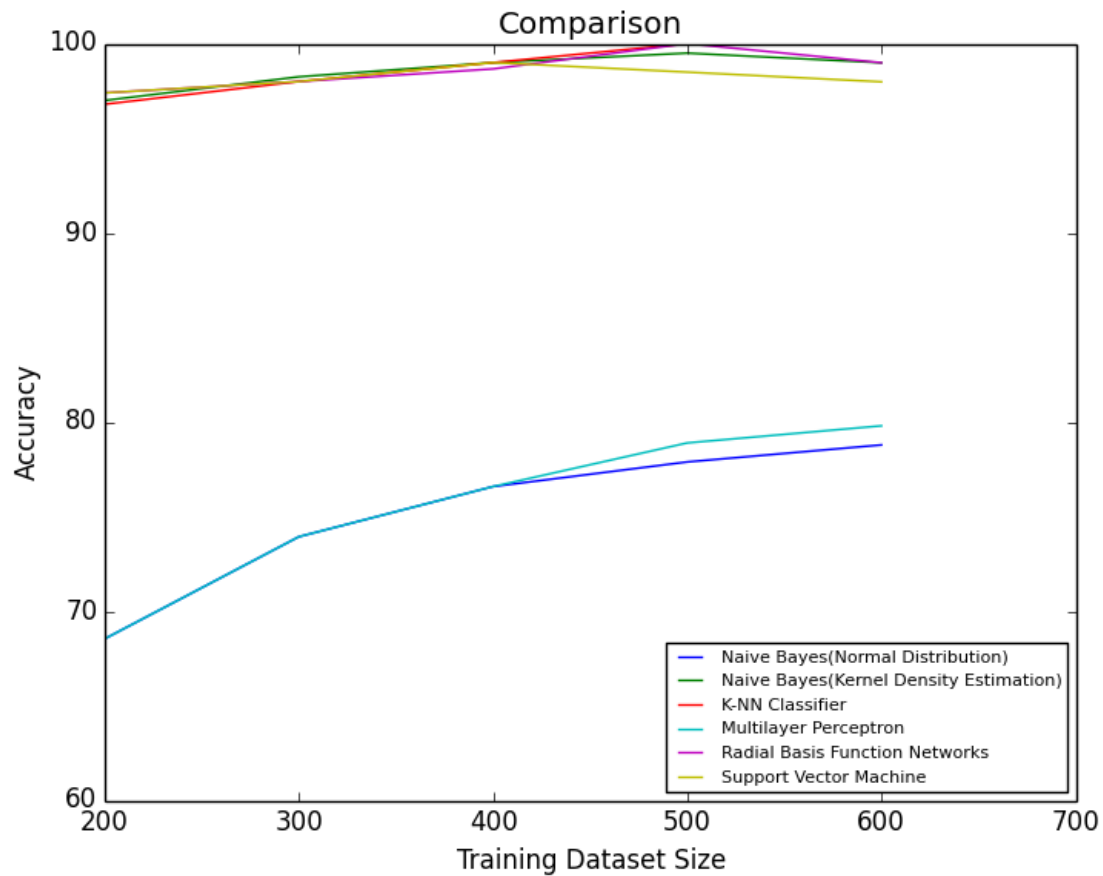


Figure 7.5: Comparison of the approaches used varying the training dataset sizes

Chapter 8

Conclusion and Future Work

In this thesis the discussed approaches for breast cancer diagnosis are implemented and the results are analyzed. Plots for specific approaches showed how their performance depended on various parameters and the parameter values that showed higher performance are noted for overall comparison of approaches. Naive Bayes using kernel density estimation, K-NN Classifier, Radial basis function networks and Support vector machine showed high and almost equal accuracies. Naive Bayes using normal distribution and Multilayer perceptron showed lower accuracies when compared to others. As the training dataset size increased the multilayer perceptron showed better accuracies than Naive Bayes using normal distribution. Naive Bayes using kernel density estimation, radial basis function networks and support vector machines showed slight decrease in their accuracies after reaching certain training dataset size.

The work in this thesis can be extended by considering other approaches for comparison and finally the best one can be used to build a breast cancer diagnostic system with higher performance. The feature extraction process that is not considered in this thesis can be researched to extract better features for higher performance.

Bibliography

- [1] Yoichi Murakami, Kenji Mizuguchi: Applying the Nave Bayes classifier with kernel density estimation to the prediction of protein-protein interaction sites. *Bioinformatics* 26(15): 1841-1848 (2010).
- [2] George H. John and Pat Langley. Estimating continuous distributions in Bayesian classifiers. In P. Besnard and S. Hanks, editors, *Eleventh Annual Conference on Uncertainty in Artificial Intelligence*, pages 338–345, San Francisco, 1995. Morgan Kaufmann Publishers.
- [3] Wilbert Sibanda and Philip Pretorius. Article: Novel Application of Multi-Layer Perceptrons (MLP) Neural Networks to Model HIV in South Africa using Seroprevalence Data from Antenatal Clinics. *International Journal of Computer Applications* 35(5):26-31, December 2011. Published by Foundation of Computer Science, New York, USA.
- [4] Dustin Boswell: Introduction to support vector machines.
<http://www.work.caltech.edu/boswell/IntroToSVM.pdf>.
- [5] Cristianini N. and Shawe-Taylor J. 2000. *An Introduction to Support Vector Machines*. Cambridge University Press, Cambridge, UK.
- [6] J. Park, I.W. Sandberg :Approximation and radial basis function networks .*Neural Comput*, 5 (1993), pp. 305-316.
- [7] Domingos,P.A few useful things to know about machine learning. *Commun. ACM*.55 (10):78-87 (2012).
- [8] G.D. Magoulas, A. Prentza, Machine learning in medical applications, in: G. Paliouras, V. Karkaletsis, C.D Spyropoulos (Eds.), *Machine Learning and its Applications*, Lecture Notes in Computer Science, Springer-Verlag, Berlin, 2001, pp. 300-307.

BIBLIOGRAPHY

- [9] Mousa, R., Munib, Q., Moussa, A., 2005. Breast cancer diagnosis system based on wavelet analysis and fuzzy-neural. *Expert Syst. Appl.* 28, 713-723.
- [10] Penna-Reyes, C.A., Sipper, M., 2000. A fuzzy genetic approach to breast cancer diagnosis. *Artificial Intell. Med.* 17, 131-155.
- [11] I. Kononenko, Machine learning for medical diagnosis: history, state of the art and perspective, *Artif. Intell. Med.* 23 (2001) 89109.
- [12] Gardner, M. W., and Dorling, S. R. (1998). "Artificial neural networks (The multilayer perceptron) - A review of applications in the atmospheric sciences." *Atmospheric Environment*, 32(14/15), 2627-2636.
- [13] Park J, Sandberg IW. Approximation and radial-basis-function networks. *Neural Comput* 1993; 5: 305-16.